

SMP/E Basics for the Beginner



Session 22296

Tuesday, March 13, 2018 - 3:15 PM

Thomas Conley

Pinnacle Consulting Group, Inc. (PCG)

59 Applewood Drive

Rochester, NY 14612-3501

P: (585)720-0012 / F: (585)723-3713

pincons@rochester.rr.com

William J. Smith, MA Ed

TRIDENT Services

Solutions for z/OS® & Enterprise Servers

Capitola, CA 95010-3929

(831) 465-7661

www.triserv.com



Abstract

Are you new to z/OS® systems programming? Have you been asked to install or service a product with SMP/E, the System Modification Program? Are you interested in learning more about SMP/E?

If you answered 'YES' to any of these questions, then this session is for you! The instructor has decades of experience working with SMP/E.

The thought of modifying an operating system, applying service to it, and deploying it is a daunting undertaking. Without question, using SMP/E and understanding its terminology are intimidating: ZONE, ZONEINDEX, GLOBAL, TARGET, DLIB, and FMID – just to name a few.

Let's cut through the jargon, pursue a robust overview, and establish a solid foundation in SMP/E fundamentals!



SMP/E Version 3 Publications Family

- Knowledge Center

https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.3.0/com.ibm.zos.v2r3.gim/gim.htm

- SA23-2275-30 – SMP/E for z/OS Commands
- GA32-0883-30 – SMP/E for z/OS Messages, Codes, and Diagnosis
- SA23-2276-30 – SMP/E for z/OS Reference
- SA23-2277-30 – SMP/E for z/OS User's Guide (a primer)
- Useful: Software Delivery, Standard Packaging Rules for z/OS-Based Products, SC23-3695-10: <https://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC23-3695-10>



Agenda

- What is SMP/E?
- Key Concept: SMP/E Zones
- Creating SMP/E Zones
- SMP/E Data Sets
- DDDEFs
- SMP/E Modification Control Statements: MCS
- SMP/E Elements



Agenda

- SMP/E Commands: Fundamentals
 - ❖ RECEIVE
 - ❖ CHECK Option
 - ❖ APPLY
 - ❖ ACCEPT
 - ❖ RESTORE
 - ❖ REJECT
- SMP/E Options: The GLOBAL Zone



What is SMP/E?

- "System Modification Program - Extended"
- SMP/E is the program that installs and maintains software on an IBM® z/OS® operating system.
- SMP/E also installs and maintains software for other z/OS subsystems like DB2® and CICS®, as well as OEM software from many non-IBM vendors.
- SMP/E, when properly used, enables you to *safely* and *routinely* install and service z/OS and accompanying software products with the confidence of achieving a positive outcome!



Historical Backdrop

- Before SMP/E, generating and maintaining an operating system (OS/360 MFT and MVT) was a labor-intensive, time-consuming process known as SYSGEN (system generation) similar to MAKEing a Linux® kernel.
- Applying service was error-prone: fixes were transposed from microfiche and hardcopy listings to punch cards.
- IBM developed SMP, and later SMP/E, to simplify installation, service, and support. SMP was initially offered in 1974 with OS/360 MVT Release 21 and OS/VS2 Release 2 MVS™. It replaced an OS/360 MVT service aid called IMAPTFLE.



Unique, Robust, Mature, Evolving

- SMP/E is profoundly *unique, robust, mature, and evolving*: it technically differentiates z/OS, software service, and support in a fashion that is functionally and unequivocally superior.
- Spanning 5 decades of research and development, SMP's capabilities are virtually without parallel: it's the *gold* standard.
- SMP is a "no-charge" component of z/OS with a rich set of APIs and is fully network enabled.
- Microsoft® Windows® installers (and Linux® installers like RPM) provide only a fraction of SMP/E functionality.



Key Concepts - SMP/E Zones

- SMP/E divides software maintenance into 3 zones: GLOBAL, TARGET, and DLIB
- Collectively, they are referred to as the Consolidated Software Inventory ("CSI"): a warehouse of software information
- The GLOBAL zone is primary an administrative zone for SMP/E
- A TARGET zone defines a run-time instance of the software installed into the TARGET zone
- A DLIB zone defines the Distribution LIBraries (DLIBs) for one or more TARGET zones



Key Concepts - SMP/E GLOBAL Zone

- The GLOBAL zone describes the products installed in the TARGET and DLIB zones
- It contains SYSMODs (SYS^{te}m MOD^{ifications}) in SMPPTSx data sets, HOLDDATA, and information regarding installed products and products to be installed
- RECEIVE is run against GLOBAL zone and is the first command issued in the maintenance process
- GLOBAL zone can have more than one TARGET and DLIB zone, as listed in the ZONEINDEX entries
- GLOBAL zone is stored in a VSAM KSDS



Key Concepts - SMP/E GLOBAL Zone

- Another key piece of information in the GLOBAL Zone is the (Sub)System **RE**lease ("SREL")
- The SREL is 4-character value, typically an alphabetic character followed by 3 numeric characters
- IBM uses the following SRELs:
 - DB2 --> P115
 - CICS --> C150
 - IMS --> P115
 - MVS --> Z038
 - NCP --> P004



Key Concepts - SMP/E TARGET Zone

- TARGET zone describes the implementation environment for products installed there
- Maintenance is applied to libraries and USS filesystems defined in the target zone
- Libraries and USS filesystems (or copies created from them) will be used to run z/OS
- APPLY and RESTORE are primary SMP/E commands run against the TARGET zone
- TARGET zone stored in separate VSAM KSDS (recommended) or embedded in a GLOBAL KSDS



Key Concepts - SMP/E DLIB Zone

- DLIB zone is used to set a "base" level of z/OS maintenance
- Allows TARGET zone to be reset to known good software level, via RESTORE command
- DLIB zone is associated with only one GLOBAL zone and one or more TARGET zones
- ACCEPT is the primary SMP/E command run against the DLIB zone
- DLIB zone is stored in separate VSAM KSDS (recommended) or embedded in the GLOBAL KSDS

Global, Target, and DLIB CSI Data Sets

```
zPDT
File Edit Font Transfer Macro Options Window Help

Menu Options View Utilities Compilers Help
-----
DSLIST - Data Sets Matching MVS.*.CSI                               Row 1 of 9
Command ==>                                          Scroll ==> PAGE

Command - Enter "/" to select action                                Message                                Volume
-----
MVS.GLOBAL.CSI                                                    *VSAM*
MVS.GLOBAL.CSI.DATA                                              B2DIS1
MVS.GLOBAL.CSI.INDEX                                             B2DIS1
MVS.MVSD.CSI                                                    *VSAM*
MVS.MVSD.CSI.DATA                                              B2DIS1
MVS.MVSD.CSI.INDEX                                             B2DIS1
MVS.MVST.CSI                                                    *VSAM*
MVS.MVST.CSI.DATA                                              B2DIS1
MVS.MVST.CSI.INDEX                                             B2DIS1
***** End of Data Set list *****

M 0.5 a 4,15
```

Key Concept - SMP/E DLIB Zone

```
zPDT
File Edit Font Transfer Macro Options Window Help

GLOBAL ZONE DEFINITION - ZONE INDEX                                Row 1 to 4 of 4
===> █                                                            SCROLL ===> PAGE

The ZONE INDEX list in the GLOBAL zone is a list of DLIB and target
zones. Verify or edit the ZONE INDEX list associated with this zone.
(Select ONE at a time. Only the first of multiple selections will be
processed.)

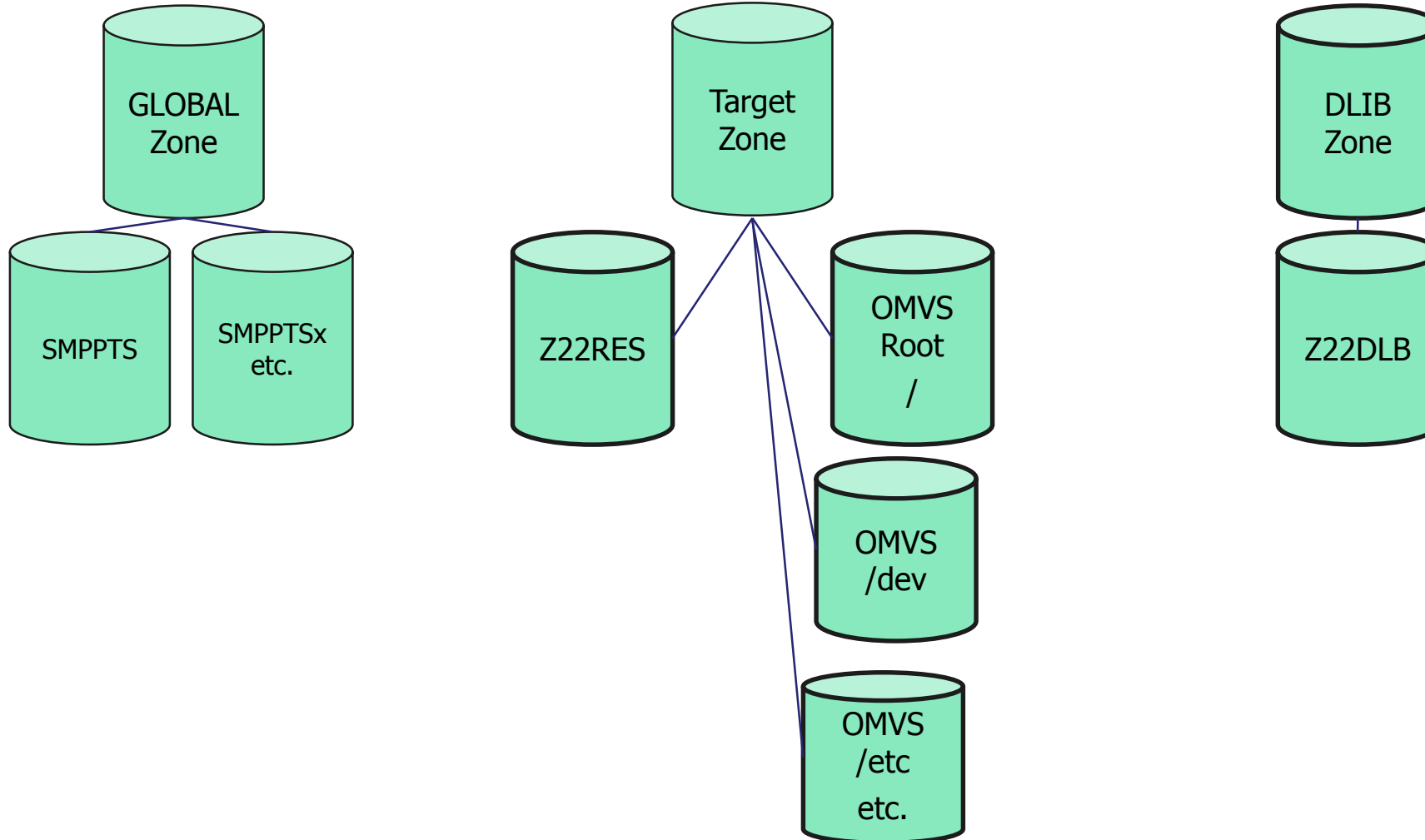
Note: Changes you make to a ZONE INDEX do not affect the associated
zone.

A - on the command line to add an entry
D - next to entry to delete an entry
S - next to entry to update an entry
When the list is complete, enter END .

OPTION  NAME  ----- CSI DATA SET NAME -----  TYPE
        CSQ800D  CSQ800.CSQ800D.CSI                DLIB
        CSQ800T  CSQ800.CSQ800T.CSI                TARGET
        MVSD     MVS.MVSD.CSI                      DLIB
        MVST     MVS.MVST.CSI                      TARGET
***** Bottom of data *****

MAB 0.1 a 2,7
```

Key Concept - SMP/E Zones





Creating SMP/E Zones

- For IBM products, Chapter 6 of a product's *Program Directory* typically contains information for allocating its SMP/E zones
- Sample jobs for allocating zones, target libraries, and DLIBs are usually in one of the relative files ("RELFILES") delivered with the product installation files
- Vendors supporting SMP/E installs also provide jobs for allocating and configuring the SMP/E environment



SMP/E Data Sets - GLOBAL Zone

- SMP/E requires *many* data sets to function. In general, most of them are configured to be dynamically allocated eliminating use of JCL DD statements.
- The IBM Program Directory or product installation documentation will include JCL to allocate data sets for the GLOBAL, TARGET, and DLIB zones.
- Important GLOBAL zone data sets . . .
 - The SMPPTS is the **PTF Temporary Store** (PTS for short), a library which holds products and maintenance for installation.
 - Some IBM maintenance can exceed the size of a PDS or PDSE.
 - SMPPTS1-SMPPTS99 are known as PTS “spill data sets”; they are allocated by the SMP/E administrator as needed to accommodate an increased volume of additional products and maintenance.



SMP/E Data Sets - TARGET Zone

- Important TARGET zone data sets
 - SMPLTS is Load Temporary Store, a library which holds load modules affected by maintenance.
 - SMPMTS is Macro Temporary Store, a library which holds macros affected by maintenance.
 - SMPSCDS is a temporary data set with stuff no one really understands, and funny member names like A\=3pb.[
 - SMPSTS is Source Temporary Store, a library which holds source modules affected by maintenance.
- Important: Keep these data sets intact when cloning target zones.



SMP/E Data Sets - TARGET Zone

- TARGET data sets are also defined in the TARGET zone
- TARGET data sets typically follow a naming convention in which a 'S' is used as the first letter in the final data set name qualifier:
 - <HLQ>.S<qualifier>
 - e.g. TARGET - SYS1.SBPXEXEC, SYS1.SHASSRC, etc.
- There are exceptions (typically control program data sets) such as SYS1.HELP, SYS1.LINKLIB, SYS1.LPALIB, etc.



SMP/E Data Sets - DLIB Zone

- DLIB zone data sets
 - SMPLTS, SMPMTS, SMPSCDS, SMPSTS, SMPPTSx defined in the DLIB zone
 - ACCEPT processing will optionally clean up those data sets
 - SMPPTS entries will be deleted based on PURGE option
- DLIB data sets are also defined in the DLIB zone.
- DLIB data sets typically follow a naming convention that uses an 'A' as first letter in its final data set name qualifier:
 - <HLQ>.A<qualifier>
 - e.g. TARGET - SYS1.MACLIB, DLIB - SYS1.AMACLIB; TARGET - ISP.SISPPENU, DLIB - ISP.AISPPENU
- There are many exceptions: e.g. SYS1.HHRFPENU (RACF)



SMP/E DDDEFS

- DDDEFS are **DD DEF**initions used to define data sets in a GLOBAL, TARGET, or DLIB zone. They use dynamic allocation (SVC 99).
- DDDEFS allow you to run SMP/E without using large JCL PROCs to hold DD statements for each SMP/E data set.
- DDDEFS are strongly recommended instead of PROCs.
- Unfortunately, some products ship only PROCs, so you may have to code your own DDDEFS.
- PROCs are usable for products with a few data sets; for z/OS - FUHGEDDABOTIT!



SMP/E DDDEFS

- Many DDDEFS are just required for SMP/E and various utilities to correctly function; some are optional and infrequently used/required
- SYSUT1, 2, and 3 are scratch data sets used by SMP/E for execution of various z/OS programs and utilities: e.g. ASMA90, IEWL, etc.
- SYSPRINT, SYSPUNCH are other examples of utility data sets.
- SMPWRK1-4,6 are work data sets for various SMP/E tasks (What happened to SMPWRK5?)
- You are strongly encouraged to examine the requirements for DDDEFS in the SMP/E Reference, Chapter 4. *SMP/E data sets and files.*

SMP/E DDDEFS - GLOBAL Zone

```
zPDT
File Edit Font Transfer Macro Options Window Help

DDDEF ENTRY SMPPTS - LIBRARY TYPE

===> █

Enter Library DDDEF data to allocate DD statements for
data sets to be dynamically allocated during SMP/E
processing. Values must conform to JCL conventions.
However, no parenthesis can be entered.

DATA SET NAME    ===> 'MVS.GLOBAL.SMPPTS'
                  (data set name, maximum 44 characters)
INITIAL DISP     ===> SHR      (OLD,SHR,MOD,NEW)
FINAL DISP      ===>          (KEEP,DELETE,CATALOG)
UNIT            ===>          (unit type if not cataloged)
VOLUME          ===>          (volume serial)
SPACE UNITS     ===>          (TRK, CYL, or block length)
PRIMARY         ===>          (primary space)
SECONDARY       ===>          (secondary space)
DIR             ===>          (Number of directory blocks)
SYSOUT          ===>          (SYSOUT class)
WAITFORDSN      ===> NO      (YES or NO)
PROTECT         ===> NO      (YES or NO)
SMS OPTIONS     ===> NO      (YES or NO to edit SMS Options)

Press ENTER to save the changes.
```

MA 0.0 a 2,7

SMP/E DDDEFs - TARGET Zone

```
zPDT
File Edit Font Transfer Macro Options Window Help

DEFINITION - DDDEF                                     Row 667 to 672 of 1,353
                                                         SCROLL ==> PAGE

==>

Global CSI: MVS.GLOBAL.CSI
Zone name: MVST

Verify or edit the DDDEF list. (Select ONE at a time. Only
the first of multiple selections will be processed.)

    A - on the command line to add an entry
    E - on the command line to edit the library type DDDEFs

    D - next to entry to delete an entry
    S - next to entry to update an entry

When the list is complete, enter END .

OPTION  NAME
  S  LINKLIB
      LOADLIB
      LPALIB
      MACLIB
      MIGLIB
      MODGEN

MAB 0.2 a 19,6
```

SMP/E DDDEFS - TARGET Zone

```
zPDT
File Edit Font Transfer Macro Options Window Help

DDDEF ENTRY LINKLIB - LIBRARY TYPE

===> █

Enter Library DDDEF data to allocate DD statements for
data sets to be dynamically allocated during SMP/E
processing. Values must conform to JCL conventions.
However, no parenthesis can be entered.

DATA SET NAME   ===> 'SYS1.LINKLIB'
                  (data set name, maximum 44 characters)
INITIAL DISP    ===> SHR      (OLD,SHR,MOD,NEW)
FINAL DISP      ===>          (KEEP,DELETE,CATALOG)
UNIT            ===> 3390     (unit type if not cataloged)
VOLUME          ===> Z22RES   (volume serial)
SPACE UNITS     ===>          (TRK, CYL, or block length)
PRIMARY         ===>          (primary space)
SECONDARY       ===>          (secondary space)
DIR             ===>          (Number of directory blocks)
SYSOUT          ===>          (SYSOUT class)
WAITFORDSN      ===> YES      (YES or NO)
PROTECT         ===> NO       (YES or NO)
SMS OPTIONS     ===> NO       (YES or NO to edit SMS Options)

Press ENTER to save the changes.
```

SMP/E DDDEFS - DLIB Zone

```
zPDT
File Edit Font Transfer Macro Options Window Help

DDDEF ENTRY ALINKLIB - LIBRARY TYPE

===>

Enter Library DDDEF data to allocate DD statements for
data sets to be dynamically allocated during SMP/E
processing. Values must conform to JCL conventions.
However, no parenthesis can be entered.

DATA SET NAME    ===> 'SYS1.ALINKLIB'
                  (data set name, maximum 44 characters)
INITIAL DISP     ===> SHR      (OLD,SHR,MOD,NEW)
FINAL DISP       ===>         (KEEP,DELETE,CATALOG)
UNIT             ===>         (unit type if not cataloged)
VOLUME           ===>         (volume serial)
SPACE UNITS      ===>         (TRK, CYL, or block length)
PRIMARY          ===>         (primary space)
SECONDARY        ===>         (secondary space)
DIR              ===>         (Number of directory blocks)
SYSOUT           ===>         (SYSOUT class)
WAITFORDSN       ===> YES     (YES or NO)
PROTECT          ===> NO      (YES or NO)
SMS OPTIONS      ===> NO      (YES or NO to edit SMS Options)

Press ENTER to save the changes.
```



SMP/E DDDEFS

- While SMP/E panels are OK to display or modify one or two DDDEFS, large numbers of DDDEFS will require **U**ppdate **C**ontrol **L**anguage (UCLIN).
- UCLIN can define any element to SMP/E with batch statements which you can run in bulk.
- IBM and software vendors *typically* provide JCL to create DDDEFS.
- Unfortunately, some vendors don't: you may have to translate their SMPPROC to DDDEFS.



SMP/E DDDEFS

- Why do I need DDDEFS again?
- DDDEFS make cloning target zones much easier.
- The DDDEF report shows clearly which data sets were affected.



SMP/E Modification Control Statements - MCS

- Once zones, data sets, and DDDEFs are defined, SMP/E elements defined with **M**odification **C**ontrol **S**tatements (MCS) can be installed.
- SYSMOD MCS describes products and maintenance to be installed.
- **++FUNCTION** is the main MCS for a product. Function SYSMODs are classified as either **base** or **dependent**.
- **++FUNCTION** creates and defines the **F**unction **M**odification **I**Dentifier (FMID).
- Once a base function and/or dependent function SYSMOD(s) is/are applied, maintenance is typically applied thereafter: **++APARs** and/or **++PTFs**.



SMP/E Modification Control Statements - MCS

- SYSMOD (SYS^{te}m MODⁱfication) MCS
 - ++FUNCTION - defines a product (function SYSMOD) and FMID
 - ++FEATURE - collection of related FMID's
 - ++APAR - Authorized Program Analysis Report, typically a *temporary* fix until a ++PTF is created
 - ++PTF - Program Temporary Fix (>99% of all applied maintenance!)
 - ++USERMOD - USER MODification (advanced SMP/E)
 - ++HOLD - HOLDDATA indicates special conditions for maintenance such as ACTION, IPL, ERROR, DOC, etc.
- These are the most common statements; there are others.

SMP/E Modification Control Statements - MCS

```
zPDT
File Edit Font Transfer Macro Options Window Help

Menu Utilities Compilers Help

BROWSE      MVS.GLOBAL.SMPPTS1(KA48886)      Line 0000000000 Col 001 080
Command ==>      Scroll ==> CSR
***** Top of Data *****
++ APAR(KA48886) /* 5752SC1C5 */ REWORK(2015288).
++ VER (Z038)
   FMID(HBB77A0)
   PRE  (KA47926,UA78544,KA47824,UA78789,KA48458,UA78962,
        KA47062,UA78508,KA47437)
   SUP  (AA48886).
++MOD (IEAVNPAL) CSECT(IEAVNPAL) DISTLIB(AOSB3)
   LE Parm(LET,LIST,NCAL,XREF).
$$$GIMC.....h{Wx1≠...Aln-Êcï....°...lnT8W..Àϕ)=\..6l0≡c}.g≤B.. Lq.b4-3-...5w.)3.
s|M..h≤..W<±3.ÂdIØ+ '&áĩñ°guϕ..À...→!Á\..10..n.V. -.Δ-a□...o....||&.a.BØn.Èϕb....
e\...\{Â. .a..\È¶.\Uoh0.&<..Ë.-.ϕ2.}≡~..Øj.H.a+. fJà.....K ..n:...ϕ..$≡{Â. ≡c
}. xn0.â|▽|.e'á.....İ...z\..h.....Ç\..°...□.Wk...°G|+-...K ..-...{ .+c3L0.Ë°'D.,
âI..±Ø..°%.L|..P!≠Ui.≠KNh9n)■.3ig'.°x≥i°E...:Γ~.≡zn..D|ñ¶.U.T1ϕT:POϕMΔ^..γN2\,c<.
.X..Ë`7≠...`LVT,U.
++MOD (IEFDB410) CSECT(IEFDB410) DISTLIB(AOSB3)
   LE Parm(LET,LIST,NCAL,REFR,RENT,XREF).
$$$GIMC.....Ã...†{İ~+|... ln{Â!-vAP.....n°Ã.Pi□+{.AEq.;ÈU±$Ø.....+...{□3-bã...3.,D
4.-.....L|ϕ..Lm..\IB-...^NJ...~ØD5"D≤.)f!h.ØF...7.&-...~ØD5İ.a†+&Â$.o●...iΔ.)f
..N\F...7.&-..c.~ØD5"≡D≤.)f..N6)....>..|:.....2$ +-..À$.M 4..]à-B¶Cl_-x.-ñ..n{Â!
```




SMP/E Modification Control Statements - MCS

- Explanation of the preceding slide:
 - ++APAR is KA48886, with REWORK
 - ++VER verifies SREL Z038 and FMID HBB77A0 (z/OS V2R2) with many **PRE**requisites; **SUP**ersedes AA48886
 - ++MOD shows object module IEAVNPAL, the LEPARMS for binding the module, and \$\$GIMC indicates SMP/E compressed data
 - Maintenance from IBM and ISV's is frequently compressed to save space in the SMPPTSx data set(s)
 - PDS command (file 182 - <http://cbttape.org>) includes utility **PTSEXP** (assembler source calling GIMCPTS to display SMP/E compressed data)



SMP/E Modification Control Statements - MCS

- There are many different types of modification control statements. Some are quite specialized.
- Data element MCS defines elements that are not macros (++MAC), modules (++MOD), or source (++SRC).
- Among the 27 different types of data element MCS, these are some of the most frequently used:
 - ++CLIST
 - ++EXEC
 - ++MSGxxx (when present, xxx is a language like ENU, DEU, JPN, etc.)
 - ++PROC
 - ++SAMPxxx
 - ++SKLxxx - ISPF skeletons
 - ++TBLxxx - ISPF tables



SMP/E Commands - RECEIVE

- SMP/E installations begin with the RECEIVE command
- RECEIVE loads products and maintenance into the GLOBAL zone and on disk
- By default, RECEIVE will pull PTF's and HOLDDATA from DDDEF SMPPTFIN
- HOLDDATA also in DDDEF SMPHOLD
- But that's *so* 20th century...
- Today, RECEIVE is usually followed by FROMNETWORK, FROMNTS, or ORDER



SMP/E Commands - RECEIVE

- RECEIVE FROMNETWORK will RECEIVE a GIMZIP package from a SMP/E RECEIVE server on an IP network.
- GIMZIP is a SMP/E utility program that compresses an installation package into a manageably-sized, portable file for network file transfer.



SMP/E Commands - RECEIVE

- RECEIVE FROMNTS will RECEIVE the package from DDDEF SMPNTS, which is a PATH statement for a USS filesystem directory.
- RECEIVE ORDER will RECEIVE the package directly from IBM Shopz:
 - The bitmap of current service sent to IBM and the order is *custom tailored* for your software configuration based upon the bitmap!
 - The most popular and convenient way to RECEIVE SMP/E maintenance from IBM
 - Designed to support most proxies and firewalls; there's *no* excuse not to implement!



SMP/E Commands - RECEIVE

- The only remaining operand to specify on RECEIVE is SOURCEID
- SOURCEID is a user-specifiable name that uniquely identifies maintenance to be installed
- A SOURCEID consists of a string of 1-to-64 NONBLANK characters (formerly, only 8 uppercase characters in length)
- Examples: JES2_ABEND0C4_HASPPARM, ISPF_EDIT_ENHANCEMENTS, etc.

Note: IBM creates its own uniquely named SOURCEIDs using ++ASSIGN MCS statements (e.g. RSU1712, HIPER, etc.).



SMP/E Commands - RECEIVE

- Examples.
- Notice the use of periods and continued (inline/imbedded) comment statements. The rules for coding are rigorous.

```
SET      BDY (GLOBAL)          /* Process global zone.    */.  
RECEIVE  S (UA00000,UA11111) /* Receive selected      */  
        SYSMODS                /* SYSMODs                */  
        HOLDDATA                /* and HOLDDATA applicable  
                                to them,          */  
        SOURCEID (SDSFFIX) /* and assign a SOURCEID. */.
```

SMP/E Commands - RECEIVE

■ Examples (continued). . . .

```
//RECEIVE EXEC PGM=GIMSMP
//SMPCSI DD DSN=SMPE.GLOBAL.CSI,DISP=SHR
//SMPNTS DD PATH='/u/smpe/smpnts/',PATHDISP=KEEP
//SMPOUT DD SYSOUT=*
//SMPRPT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPCNTL DD *
SET BOUNDARY(GLOBAL) .
RECEIVE SYSMODS HOLDDATA
    ORDER(
        PENDING(ORD00035) /* Get the specified pending order */
        CLIENT(MYCLIENT)
    ) .
/*
//MYCLIENT DD*
<CLIENT
    javahome="/usr/lpp/java/J1.4"
    classpath="/usr/lpp/smp/classes">
</CLIENT>
/*
```




SMP/E Commands - CHECK Option

- APPLY, ACCEPT, RESTORE, and REJECT
- The CHECK operand may be optionally appended to each of these commands to execute a “dry run” – i.e. a simulation - without updating the target and distribution data sets
 - **ALWAYS ... Run APPLY CHECK before APPLY**
 - **ALWAYS ... Run ACCEPT CHECK before ACCEPT**
 - **ALWAYS ... Run RESTORE CHECK before RESTORE**
 - **ALWAYS ... Run REJECT CHECK before REJECT**



SMP/E Commands - CHECK Option

- Why waste my time with CHECK? I never make any SMP/E errors!
- Yeah, right - until you do - *and will*
- CHECK will enable you to verify the following:
 - Did/will all of the correct maintenance install? Are there missing pre-requisites?
 - Did/will the service install in the correct place – i.e. data sets and paths? (ALWAYS inspect the DDDEF report to ensure you're updating correct libraries.)
 - Is there HOLDDATA that informs you to do something before the APPLY? (A common HOLD ACTION is to increase the size/directory blocks of a library before APPLY to avoid X37 ABENDs.)



SMP/E Commands - APPLY

- APPLY installs products and maintenance into TARGET zone
- Important part of APPLY processing is resolving HOLDDATA
- Sample of SMP/E messages for HOLDDATA:

GIM42001I THE FOLLOWING CONDITIONS FOR SYSMOD UA79029 WERE NOT SATISFIED, BUT
WERE IGNORED BECAUSE THE BYPASS OPERAND WAS SPECIFIED. PROCESSING
CONTINUES.

GIM35966I SYSTEM HOLD IPL ORIGINATED BY SYSMOD UA79029 WAS BYPASSED.

GIM35966I SYSTEM HOLD DOC ORIGINATED BY SYSMOD UA79029 WAS BYPASSED.

GIM35966I SYSTEM HOLD ENH ORIGINATED BY SYSMOD UA79029 WAS BYPASSED.

GIM35966I SYSTEM HOLD ACTION ORIGINATED BY SYSMOD UA79029 WAS BYPASSED.

- Shows four HOLDS of four different types for UA79029



SMP/E Commands - APPLY

■ ++ HOLDS in SMPPTS(UA79029)

```
++ HOLD(UA79029) SYS FMID(HBB7790) REASON(IPL) DATE(15259)
```

```
COMMENT
```

```
(*****
```

```
* FUNCTION AFFECTED: 5752SC1CR (OA47439) *
```

```
* REAL STOR MGR 790 *
```

```
*****
```

```
* DESCRIPTION : IPL with CLPA *
```

```
*****
```

```
* TIMING : Post-APPLY *
```

```
*****
```

```
In order for this PTF to be fully effective, an IPL with CLPA is  
required.
```

SMP/E Commands - APPLY

■ ++ HOLDS in SMPPTS(UA79029)

```
++ HOLD(UA79029) SYS FMID(HBB7790) REASON(DOC) DATE(15259)
```

```
COMMENT
```

```
(*****  
* FUNCTION AFFECTED: Real Storage Mgr                (OA47439) *  
*                      Real Storage Mgr                *  
*****  
* DESCRIPTION      : Documentation updates                *  
*****  
* TIMING           : As Needed                            *  
*****  
1.      SA23-1370-01  z/OS MVS Programming: Assembler  
                      Services Reference IAR-XCT  
2.      SA23-1373-01  z/OS MVS Programming: Authorized  
                      Assembler Services Reference EDT-IXG
```

The following parameter is added to the IARV64 Macro for
REQUEST=GETSTOR, REQUEST=GETCOMMON, and REQUEST=GETSHARED: ...



SMP/E Commands - APPLY

■ ++ HOLDS in SMPPTS(UA79029)

```
++ HOLD(UA79029) SYS FMID(HBB7790) REASON(ENH) DATE(15259)
```

```
COMMENT
```

```
(*****
```

```
* FUNCTION AFFECTED: Real Storage Mgr (OA47439) *
```

```
* Real Storage Mgr *
```

```
*****
```

```
* DESCRIPTION : Enhancement *
```

```
*****
```

```
* TIMING : As Needed *
```

```
*****
```

```
This APAR provides new function to support customers running  
with large amounts of real storage, usually between 1-4 TB.).
```



SMP/E Commands - APPLY

- ++ HOLDS in SMPPTS(UA79029)

```
++ HOLD (UA79029) SYS FMID (HBB7790) REASON (ACTION) DATE (15259)  
COMMENT
```

(The Stand-Alone Dump program must be rebuilt after this PTF is applied.) .

- After applying the PTF, you must run a job to rebuild SADUMP [z/OS MVS Diagnosis: Tools and Service Aids](#); this is a typical post-APPLY practice.
- Satisfy **all** ++HOLD entries in APPLY CHECK output before proceeding to APPLY.



SMP/E Commands - APPLY

- Many APPLY operands are "adults only". . .
 - ***DO NOT use the following operands unless you know what you're doing and/or unless directed otherwise by IBM or an ISV!***
 - ***BYPASS(HOLDSYSTEM) is the only BYPASS operand you need or want. The others are potentially dangerous!***
- Bypassing SYSTEM HOLDS creates a HOLDDATA report
- <Tell the cautionary tale of IBMainiac who ran
BYPASS(HOLDSYS,HOLDERR,ID)>



SMP/E Commands - APPLY

- **"REDO" IS DECEPTIVELY DANGEROUS, DON'T USE IT!**
 - Many believe one should specify REDO after an SMP/E APPLY or ACCEPT fails. That choice is INCORRECT. DON'T DO IT!
 - When an SMP/E APPLY/ACCEPT fails, SMP/E will AUTOMATICALLY continue from where it left off.
 - Just restart APPLY/ACCEPT. Let SMP/E do "its thing"!
 - Why is REDO dangerous?
 - ++JCLIN statements are cumulative, so REDO will attempt to re-APPLY ++JCLIN, potentially creating errors
- **JUST SAY *NO* TO "REDO"**



SMP/E Commands - APPLY

- **"REUSE" IS RIDICULOUSLY DANGEROUS. DON'T USE IT!**
 - REUSE is unquestionably *the* most dangerous APPLY operand.
 - Everything must be perfect in order to use it.
 - REUSE is a relic of the 1980's - when DASD and CPU were expensive.
 - By today's standards, DASD and CPU are relatively inexpensive and plentiful commodity items when compared to the damage that can be done by REUSE.
- **JUST SAY *NO* TO "REUSE"**



SMP/E Commands - ACCEPT

- ACCEPT installs products and maintenance into a DLIB zone to be used as a baseline.
- ACCEPT creates a fully functional level-set of a product or maintenance.
- RESTORE processing will use this level-set from ACCEPTed maintenance to return the system back to a functional, working state.
- All of the APPLY items discussed earlier are also pertinent to ACCEPT (HOLDDATA, etc.).
- **ACCEPT ONLY ++PTFs**
- **>NEVER< ACCEPT ++APARs or ++USERMODs**



SMP/E Commands - ACCEPT

- ACCEPT myth (*busted*)

*"You can't recover if you ACCEPT maintenance that later fails.
Therefore, never ACCEPT."*

- Backup your entire SMP/E environment and restore the backups if you have to recover to a point in time prior to ACCEPT.
- If you ever have to restore a PTF, and if you haven't ACCEPTed anything, RESTORE will either fail outright, or RESTORE everything, including the ++FUNCTION.



SMP/E Commands - RESTORE

- RESTORE uninstalls products and/or maintenance in the TARGET zone.
- DLIB copies of elements affected by RESTORE will replace TARGET zone elements.

For example: If RESTOREd maintenance affects the CLOSE macro, then SYS1.AMACLIB(CLOSE) will be copied to SYS1.MACLIB(CLOSE).

- Resets element(s) to a functionally acceptable working state



SMP/E Commands - REJECT

- REJECT will delete SYSMOD or HOLDDATA from the GLOBAL zone and the SMPPTSx data set(s).
- REJECT usually takes place when ACCEPTing maintenance and the SMP/E option is to REJECT.
- The REJECT command can also be issued to purge (i.e. 'reject') individual PTF's *in error* in order to prepare for them to be re-RECEIVED.



SMP/E Options - GLOBAL Zone

- GLOBAL zone defines many SMP/E processing options.
- The GLOBAL zone defaults are appropriate for most installations.
- However, here are two prudent options to review:
 1. **PURGE** - Do you want to delete a SYSMOD from the SMPPTS after ACCEPT processing? NOPURGE will not delete the SYSMOD, so your SMPPTS data set keeps growing, and growing.
 2. **REJECT** - Do you want to REJECT a SYSMOD after it's been RESTOREd? The assumption is a SYSMOD was somehow in error and needs to be RECEIVED again. Therefore, REJECTing it on RESTORE may save time.



Finally...

Please complete a session evaluation.

Your comments help deliver better presentations.

I'm interested in hearing about your experiences with SMP/E; if you encounter any issues and would like assistance, send an email to pincons@rochester.rr.com